**SSN College of Engineering, Kalavakkam**

**Department of Computer Science and Engineering**

**III Semester**

**UCS 1312 Data Structures Lab Laboratory**

**Academic Year: 2019-2020 Batch: 2018-2022**

**Name: S. Prathyush**

**Class: CSE Sec: B**

**Roll No:- 185001112**

**ASSIGNMENT – 10:PRIORITY QUEUE USING BINARY HEAP**

**Header file - Employee.h**

typedef struct Employee{

char name[30];

int id;

float salary;

}Employee;

Employee getEmployee(){

Employee e;

printf("Enter the name : ");

scanf("%[^\n]",e.name);

printf("Enter the id : ");

scanf("%d",&e.id);

printf("Enter the salary : ");

scanf("%f",&e.salary);

getchar();

printf("\n");

return e;

}

void putEmployee(const Employee e){

printf("Name : %s\n",e.name);

printf("ID : %d\n",e.id);

printf("Salary : %.2f\n",e.salary);

}

-------------------------------------------------------------------------------------------------------

**Header file - MaxHeap.h**

typedef Employee Data;

typedef struct PriorityQueue{

int capacity;

int size;

Data\* arr;

}PriorityQueue;

typedef PriorityQueue\* PQueue;

int isFull(PQueue Q){

return Q -> size == Q -> capacity;

}

int isEmpty(PQueue Q){

return Q -> size == 0;

}

PQueue createPQueue(const int maxsize){

PQueue tmp = (PQueue)malloc(sizeof(PriorityQueue));

tmp -> capacity = maxsize;

tmp -> size = 0;

tmp -> arr = (Data\*)malloc(sizeof(Data) \* maxsize);

tmp -> arr[0].salary = 999999.9;

return tmp;

}

void enqueue(PQueue q,const Data d){

if(isFull(q)){

printf("Queue Full!\n");

return;

}

int i = ++q -> size;

for(; q -> arr[i/2].salary < d.salary ; i /= 2)

q -> arr[i] = q -> arr[i/2];

q -> arr[i] = d;

}

Data dequeue(PQueue q){

if(isEmpty(q)){

printf("Queue Empty!\n");

return q -> arr[0];

}

int i,child;

Data min,last;

min = q -> arr[1];

last = q -> arr[q -> size--];

for(i = 1; i \* 2 <= q -> size ; i = child){

child = i \* 2;

if(child != q -> size && q -> arr[child + 1].salary > q -> arr[child].salary)

child ++;

if(last.salary < q -> arr[child].salary)

q -> arr[i] = q -> arr[child];

else

break;

}

q -> arr[i] = last;

return min;

}

void display(PQueue Q){

for(int i = 1 ; i <= Q -> size ; i++)

putEmployee(Q -> arr[i]);

}

-------------------------------------------------------------------------------------------------------

**Main.c**

#include <stdio.h>

#include <stdlib.h>

#include "Employee.h"

#include "MaxHeap.h"

int main(void){

PQueue q = createPQueue(10);

for(int i = 0 ; i < 5 ; i++){

enqueue(q,getEmployee());

printf("Queue after adding: \n");

display(q);

printf("------------------------\n");

}

printf("De-Queued Element\n");

putEmployee(dequeue(q));

}

**OUTPUT:**

(base) Prathyush@Prathyush-PC:/Assignment-10$ ./main

Enter the name : Prathyush

Enter the id : 1

Enter the salary : 26500

Queue after adding:

Name : Prathyush

ID : 1

Salary : 26500.00

------------------------

Enter the name : Praveen

Enter the id : 2

Enter the salary : 12000

Queue after adding:

Name : Prathyush

ID : 1

Salary : 26500.00

Name : Praveen

ID : 2

Salary : 12000.00

------------------------

Enter the name : Aravind

Enter the id : 3

Enter the salary : 36500

Queue after adding:

Name : Aravind

ID : 3

Salary : 36500.00

Name : Praveen

ID : 2

Salary : 12000.00

Name : Prathyush

ID : 1

Salary : 26500.00

------------------------

Enter the name : Hari

Enter the id : 4

Enter the salary : 33000

Queue after adding:

Name : Aravind

ID : 3

Salary : 36500.00

Name : Hari

ID : 4

Salary : 33000.00

Name : Prathyush

ID : 1

Salary : 26500.00

Name : Praveen

ID : 2

Salary : 12000.00

------------------------

Enter the name : Kumar

Enter the id : 5

Enter the salary : 65000

Queue after adding:

Name : Kumar

ID : 5

Salary : 65000.00

Name : Aravind

ID : 3

Salary : 36500.00

Name : Prathyush

ID : 1

Salary : 26500.00

Name : Praveen

ID : 2

Salary : 12000.00

Name : Hari

ID : 4

Salary : 33000.00

------------------------

De-Queued Element

Name : Kumar

ID : 5

Salary : 65000.00

==========================================================